

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: 06332848 A

(43) Date of publication of application: 02.12.94

(51) Int. Cl.

G06F 13/36

G06F 13/38

(21) Application number: 05117271

(71) Applicant: FUJITSU LTD

(22) Date of filing: 19.05.93

(72) Inventor: NAKAMURA YOICHI

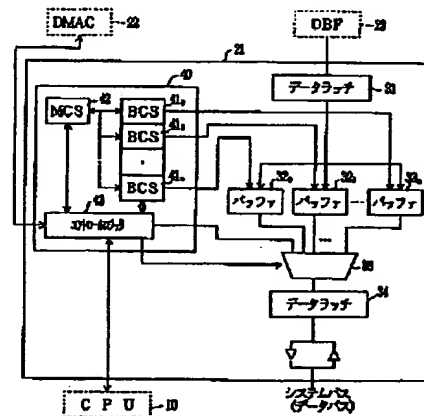
(54) DATA TRANSFER SYSTEM

COPYRIGHT: (C)1994,JPO

(57) Abstract:

PURPOSE: To accelerate data transfer, to prevent the futile use of a system bus and to improve transfer efficiency by providing plural buffer control means corresponding to plural buffers and successively transferring data for each buffer which completed transfer preparation.

CONSTITUTION: This system is a computer system for which an adapter 11 for controlling the various kinds of devices and a CPU 10 are connected by a system bus 12. The CPU 10 or the adapter 11 uses a split system for transmitting instructions and the data to an opposite party and tentatively opening the system bus 12 and transfers the data. The adapter 11 is provided with the plural buffers 32₀-32_n and the plural buffer control means 41₀-41_n corresponding to the plural buffers 32₀-32_n for successively transferring the data to the CPU 10 for each buffer when the data are respectively stored and the transfer preparation is completed. Since the data are successively transferred for each buffer by the plural buffer control means 41₀-41_n in such a manner, the data can be transferred at a high speed.



【特許請求の範囲】

【請求項1】 各種デバイスを制御するアダプタ（11）とCPU（10）とをシステムバス（12）により接続したコンピュータシステムで、CPU（10）又はアダプタ（11）は命令及びデータを相手に送信して一旦システムバス（12）を開放するスプリット方式を用いてデータを転送するデータ転送方式において、上記アダプタ（11）に、複数のバッファ（32、～32、）と、上記複数のバッファ（32、～32、）夫々にデータが蓄積され転送準備が完了するとバッファ毎に順次CPUに対してデータ転送を行なわせる上記複数のバッファ（32、～32、）に対応した複数のバッファ制御手段（41、～41、）とを有することを特徴とするデータ転送方式。

【請求項2】 請求項1記載のデータ転送方式において、前記アダプタ（11）に、複数のバッファ制御手段（41、～41、）を監視して前記複数のバッファ（32、～32、）のデータ転送が完了するまで他のバッファ制御手段による次の転送サイクルのデータ転送を待たせるマスタ制御手段（42）を有することを特徴とするデータ転送方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、CPUと各種デバイスを制御するアダプタとをシステムバスで接続したコンピュータシステムのデータ転送方式に関する。

【0002】

【従来の技術】 従来より、ソフトウェアの走行するCPUと、CPUの命令によってハードディスク装置や通信回線等のデバイスを制御するアダプタ等から構成されるコンピュータシステムにおいて、CPUとアダプタとを接続するバスはスプリット方式で制御されていた。

【0003】 スプリット方式はCPUがアダプタに対して命令を発行するときのバスオペレーションで、通常はCPUが命令を発行して結果を得るまでバスをインターロックして使用するのに対して、スプリット方式ではCPUは命令を発行した時点でバスを開放し、アダプタで結果の準備ができるまで他のCPUやアダプタがバスを使用できる。このようなコンピュータシステムにおいてソフトウェアの命令によりアダプタからCPUに対して大量のデータ転送を行なうとき、アダプタはデバイスから読込んだデータを一旦内蔵するシステムバスコントローラ内のバッファに蓄積してCPUに転送する。

【0004】

【発明が解決しようとする課題】 従来のスプリット方式のデータ転送方式では、アダプタのシステムバスコントローラからCPUにデータの転送が終了すると、CPUから正常受信したという結果が得られるまで次のデータ

転送ができなかった。

【0005】 このため、バスクロックを高速化したり、バッファの数を増加してもデータ転送の高速化には限界があった。また、CPU側の処理が手一杯でデータ受信ができないビジー状態が発生すると、アダプタはある一定時間を待って再転送つまりリトライ処理を行なうが、そこでデータ転送が滞るため転送効率が低下するという問題があった。

【0006】 本発明は上記の点に鑑みなされたもので、データ転送を高速化し、システムバスの無駄な使用を防止して転送効率を向上させるデータ転送方式を提供することを目的とする。

【0007】

【課題を解決するための手段】 本発明のデータ転送方式は、各種デバイスを制御するアダプタとCPUとをシステムバスにより接続したコンピュータシステムで、CPU又はアダプタは命令及びデータを相手に送信して一旦システムバスを開放するスプリット方式を用いてデータを転送するデータ転送方式において、上記アダプタに、複数のバッファと、上記複数のバッファ夫々にデータが蓄積され転送準備が完了するとバッファ毎に順次CPUに対してデータ転送を行なわせる上記複数のバッファに対応した複数のバッファ制御手段とを有する。

【0008】 また、前記アダプタに、複数のバッファ制御手段を監視して前記複数のバッファのデータ転送が完了するまで他のバッファ制御手段による次の転送サイクルのデータ転送を待たせるマスタ制御手段を有する。

【0009】

【作用】 本発明においては、複数のバッファ制御手段によって、バッファ毎に順次データ転送を行なわせるため、高速のデータ転送が可能となる。

【0010】 また、マスタ制御手段により全てのバッファのデータ転送が終了した後次の転送サイクルを開始させるため、任意のバッファ制御手段のデータ転送時にビジーが発生しても他のバッファ制御手段のデータ転送が上記ビジーに起因するリトライ動作が終了するまで待たされず、システムバスの無駄な使用を防止される。

【0011】

【実施例】 図2は本発明方式が適用されるコンピュータシステムの構成図を示す。同図中、CPU10はマイクロプロセッサとメモリから成りメモリ上に基本ソフトウェア（OS）を搭載して動作する。アダプタ11はディスク装置13や通信回線14等のデバイスを制御する。システムバス12はCPU10とアダプタ11を結合するスプリット方式のバスである。OSの命令によりCPU10はアダプタ11に対して命令を発行する。アダプタ11はこの命令に対してデバイスからデータを読み出したり、メモリ上のデータをデバイスに対して書き出したりする動作を行なう。

【0012】 図3はアダプタ10のブロック図を示す。

システムバスコントローラ (SBC) 21はCPU10と接続するシステムバス12を制御する。DMAコントローラ (DMAC) 22はデバイスとのDMA転送を制御する。データバッファ (DBF) 23はDMA転送で使用するデバイスインタフェースコントローラ (DIF) 24はデバイスとのインタフェースを制御する。マイクロプロセッサユニット (MPU) 25はアダプタの制御中枢である。ROM26にはMPU25の動作するプログラムが格納されている。RAM27はデータのワーク領域として用いられる。アダプタ11は、電源オンでROM26内のプログラムがRAM27上に読み出されて動作開始となる。プログラムはハードウェア各部の初期化が終了するとハードウェアの各機能の動作状態を確認し、CPU10からの命令待ち状態となる。CPU10から発行された命令はSBC21のSBC21で受信されMPU25に通知される。MPU25はこの命令を解析し、命令を実行するために各コントローラをプログラムに従って制御する。例えば、CPU10からの命令が「ディスク装置13の所定位置からデータを読み出して来る」だったとすると、MPU25はディスク装置13の所定位置からデータを読み出す命令をDIC24に発行する。DIC24はディスク装置13のデータ転送準備が完了すると、これをMPU25に通知する。MPU25はDMAC22を制御してディスク装置13から送られてくるデータをDBF23に蓄積する。DBF23にある程度データが蓄積されるとDMAC22はSBC21に対してデータ転送を開始し、SBC21は内部のバッファが一杯になった時点でシステムバス12に対してバス使用権を要求し、バス使用権が与えられるとデータをCPU10に対して送出する。CPU10はこのデータを正常に受信するとアダプタ11に対して正常受信したことを通知する。

【0013】図1は本発明方式を適用したSBC21のDMA制御部のブロック図を示す。同図中、データラッチ31はDBF23から供給されるデータを一旦ラッチする。データラッチ31の出力データはバッファ32、～32、のいずれかに蓄積され、バッファ32、～32、夫々から読出されたデータはセクタ33で選択される。セクタ33の出力データは一旦データラッチ34にラッチされ、ここからシステムバス12に送出される。

【0014】制御部40はバッファコントロールシーケンサ (BCS) 41、～41、と、マスタコントロールシーケンサ (MCS) 42と、コントロールロジック43とより構成されている。BCS41、～41、は夫々に対応するバッファ32、～32、を各別に制御し、BCS41、～41、はMCS42によって監視制御される。コントロールロジック43はDMAC22からの指示に応じてBCS41、～41、及びMCS42を制御すると共にセクタ33の制御を行なう。

【0015】BCS41、～41、夫々は図4に示す如く、3ビットで対応するバッファの動作状態を定義している。このシーケンサで定義するのは、アダプタ11からシステムバス12を通してCPU10にデータを送出する際のものである。

【0016】状態0は、アイドル状態でハードウェアは何も動作していない状態である。

【0017】状態1は、転送許可待ち状態でDBF23から読み込んだデータがバッファ上に満杯になりCPU10に対して転送するための許可を求めている状態で、システムバス12が空き次第転送が開始される。

【0018】状態2は、転送許可が与えられてデータをCPU10に向けて転送中の状態である。

【0019】状態3は、データ転送が終了し、転送結果を待っている状態である。

【0020】状態Eは、転送終了の状態である。

【0021】MCS42は図5に示す如く、3ビットでBCS40、～40、夫々の動作状態を定義している。これはBCSが4個 (BCS40、～BCS40、) の場合の例である。

【0022】状態0は、アイドル状態でハードウェアは何もしない状態である。

【0023】状態1は、BCS40、～BCS40、のうちいずれかが転送終了した状態である。

【0024】状態2は、状態1で転送終了したBCSを除く残りのBCSのうち、いずれかが転送終了した状態である。

【0025】状態3は、状態1及び状態2で終了したBCSを除く残りのBCSのうちいずれか一方が転送終了した状態である。

【0026】状態Eは、残り1つのBCSが転送終了して4つのバッファの転送がひととおり終了した状態である。

【0027】ここで、図6 (A) に示す如く、システムクロックのサイクルS0ではMCS42がアイドル状態0で、かつ図6 (B) に示す如くBCS41、がアイドル状態0 (他のBCS41、～41、も状態0) にある。サイクルS1でBCS41、に対応するバッファ32、がデータで一杯になると、コントロールロジック43はCPU10に図6 (H) に示すアダプタ11のシステムバス獲得要求信号ADP. SYS. BUS. REQを送ると共に、BCS41、は転送許可待ち状態1となる。

【0028】この後、サイクルS4で図6 (F) に示すアダプタ11のシステムバス獲得確立信号ADP. SYS. BUS. GRNが得られると、次のサイクルS5でBCS41、はデータ転送中状態2となり、コントロールロジック43は図6 (J) に示すシステムバス使用開始通知信号ADP. BUS. STARTをCPU10に送ると共に、コントロールロジック43は図6 (G) に

示す如く、DMA転送をCPU10に通知するコマンドCをセクタ33、データラッチ34を通してシステムバス12のデータバスに送出する。更にサイクルS6でコントロールロジック43はデータを格納する転送先アドレスAをセクタ33、データラッチ34を通してシステムバス12に送出する。

【0029】これに続いてコントロールロジック43はセクタ33にバッファ32。を選択させ、BCS41。はサイクルS7～S14でバッファ32。の蓄積データD0～D7をセクタ33、データラッチ34を通してシステムバス12に送出する。また、サイクルS14ではコントロールロジック43は図6(K)に示す如くシステムバス終了通信信号ADP. BUS. COMPLETEをCPU10に送る。これによって次のサイクルS15でBCS41。は転送結果確認待ち状態3となる。

【0030】この後、サイクルS17でCPU10は図6(C)に示すCPU10のシステムバス獲得要求信号CPU. SYS. BUS. REQをアダプタ11に送り、サイクルS18で図6(D)に示すCPU10のシステムバス獲得確立信号CPU. SYS. BUS. GRNが得られると、次のサイクルS19でCPU10は図6(E)に示すシステムバス使用開始通知信号CPU. BUS. STARTをアダプタ11に送り、図6(G)に示す如くデータ転送に対する応答Sをシステムバス12を通してアダプタ11に送り、更に図6(F)に示すシステムバス使用終了通知信号CPU. BUS. COMPLETEをアダプタ11に送る。

【0031】これによって、サイクルS20ではMCS42は図6(A)に示す如く、1つのバッファ転送終了状態1となり、BCS41。は図6(B)に示す如く転送終了状態Eとなる。

【0032】図7は正常動作時のBCS41。～41。とMCS42との動作状態の遷移を示す。DMAC22を起動しDBF23からSBC21にデータが転送され一杯になるとBCS40。～40。夫々は状態0から状態1となる。例えばBCS40。はシステムバス12の転送開始許可を求め、転送許可が与えられると状態1から状態2となり、データ転送を行なう。データ転送が終了すると状態2から状態3となる。

【0033】これによってシステムバス12は開放されるので例えばBCS40。等の他のBCSが状態1でシステムバス12の転送開始許可を出していれば、そのBCS40。に対して転送許可が与えられる。先のBCS40。によるデータ転送が正常に終了したと確認されるとBCS40。は状態3から状態Eとなる。

【0034】このようにしてBCS40。～40。夫々によってバッファ32。～32。の蓄積データが順に転送される。各バッファが空になり更に読込むデータがあれば、DBF23からバッファ32。～32。にデータ

が読込まれ、一杯になって転送準備ができるとBCS40。～40。は状態Eから状態1となってシステムバスの転送許可を求める。

【0035】MCS42はBCS40。～40。夫々が状態Eになる毎に状態0, 1, 2, 3と状態を更新し、BCS40。～40。が全て状態Eとなった時点でMCS42は状態Eとなり、次の転送サイクルでBCS40。～40。のいずれかが状態EとなったときにMCS42は状態1となる。

【0036】このように複数のバッファ32。～32。に転送データを蓄積し、BCS41。～41。の制御により、1つのバッファのデータ転送が終了すると、その転送結果が正常だと確認される以前に、次のバッファのデータ転送を開始することによりデータ転送の高速化が可能となる。

【0037】ところで、図8に示す如く、BCS41。が状態2でデータ転送を行ない、データ転送を終了して状態3となったとき、CPU10の処理が手一杯で転送データを受信できないビジー状態が発生すると、このビジー状態の発生が判明した時点でシステムバス12の使用権はBCS41。に移っているので、BCS41。は再びシステムバス12の使用権が与えられるまで待ち、BCS41。から上記使用権が移った時点で先に転送したデータを転送する。これがリトライ動作である。

【0038】この場合、MCS42はBCS41。がデータ転送を終了しても状態3であるため、BCS41。は次のデータ転送を待たされる。この後、上記リトライ動作が完了してBCS41。が状態Eとなり、MCS42が状態Eとなると次の転送サイクルが開始されBCS41。のデータ転送が許可される。

【0039】つまり、BCS41。のデータ転送でビジー状態が発生しても、このBCS41。によるリトライ動作が完了するまでBCS41。、41。のデータ転送が待たされないため、システムバス12の無駄な使用が防止され、高速で効率の良いデータ転送が可能となる。

【0040】

【発明の効果】上述の如く、本発明のデータ転送方式によれば、データ転送を高速化し、システムバスの無駄な使用を防止して転送効率を向上させることができ、実用上きわめて有用である。

【図面の簡単な説明】

【図1】本発明方式のDMA制御部のブロック図である。

【図2】本発明方式のコンピュータシステムの構成図である。

【図3】アダプタのブロック図である。

【図4】BCSの動作状態の定義を示す図である。

【図5】MCSの動作状態の定義を示す図である。

【図6】本発明の動作を説明するためのタイミングチャートである。

【図 7】本発明の動作を説明するためのタイミングチャートである。

【図8】本発明の動作を説明するためのタイミングチャートである。

【符号の説明】

1.0 CPU

11 アダプタ

12 システムバス

21 SBC

22 DMAC

23 DBF

3 2₀ ~ 3 2_n バッファ

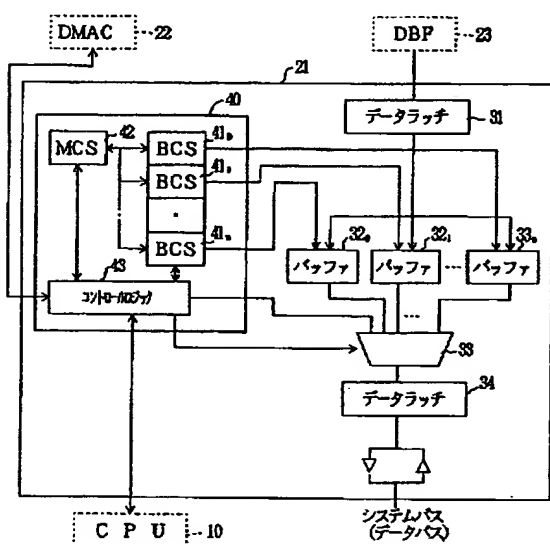
41₀ ~ 41_a BCS

42 MCS

43 コントロールロジック

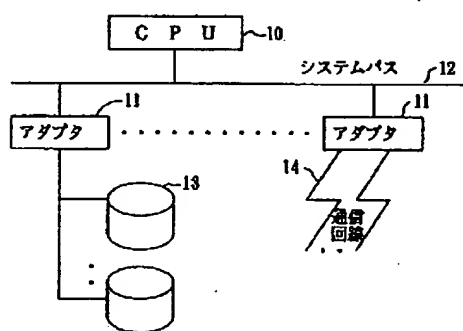
【图 1】

本発明方式のDMA制御部のブロック図



【図 2】

本発明方式のコンピュータシステムの構成図

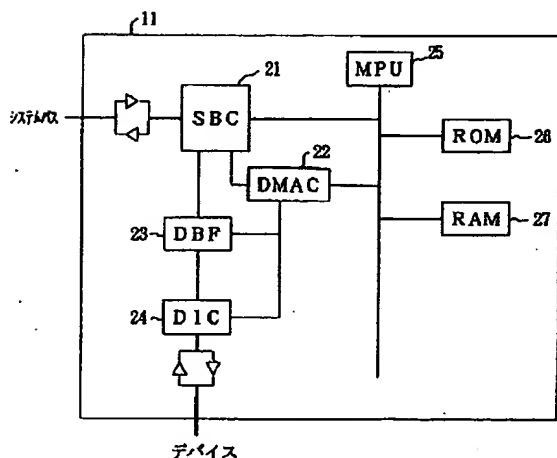


【図4】

BCSの動作状態の定義を示す図

【図 3】

アダプタのブロック図



ビット 1 2 3	シーケンサ定義	
0 0 0	0	アイドル状態
0 0 1	1	転送許可待ち状態
0 1 0	2	データ転送中
0 1 1	3	転送結果確認待ち状態
1 0 0	E	転送終了

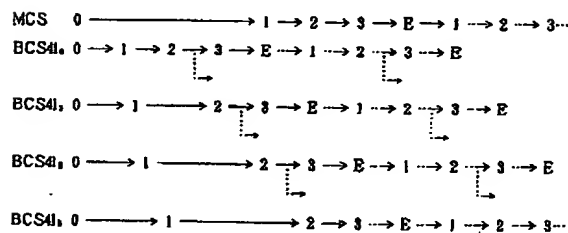
【图5】

MCSの動作状態の定義を示す図

ビット 1 2 3	シーケンサ定義	
0 0 0	0	アイドル状態
0 0 1	1	バッファx終了
0 1 0	2	バッファx終了
0 1 1	3	バッファx終了
1 0 0	E	1サイクル終了

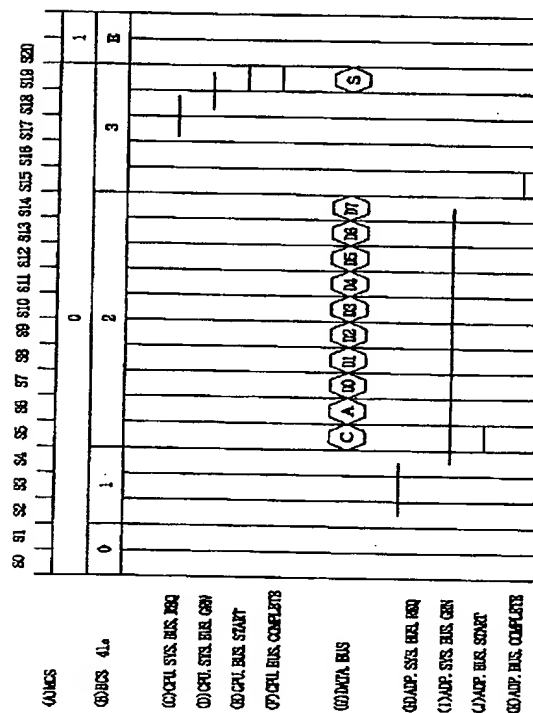
【図7】

本発明の動作を説明するためのタイミングチャート



【図 6】

本発明の動作を説明するためのタイミングチャート



【図8】

本発明の動作を説明するためのタイミングチャート

